Designing with Versal AI Engine: DSP Applications

AIE-DSP (v1.0)

Course Specification

Course Description

This course covers the AMD Versal™ AI Engine architecture and using the AI Engine DSP Library, system partitioning, rapid prototyping, and custom coding of AI Engine kernels. Developing AI Engine DSP designs using AMD Vitis™ Model Composer is also demonstrated.

The emphasis of this course is on:

- Providing an overview of the AI Engine architecture
- Utilizing the Vitis DSP library for AI Engines
- Performing system partitioning and planning
- Adding custom kernel code for designs
- Creating AI Engine DSP designs using Vitis Model Composer
- Analyzing reports using Vitis Analyzer

Level – ACAP 2

Course Details

- 1 day ILT /9

Course Part Number – AIE-DSP

Who Should Attend? – DSP users, software and hardware developers, system architects, and anyone who needs to accelerate their software applications using our devices

Prerequisites

- Comfort with the C/C++ programming language
- Vitis tool for acceleration development flow
- Comfort with basic signal processing concepts
- Basic knowledge of Versal AI Engine architecture and programming

Software Tools

- Vitis Unified IDE 2024.1
- Vitis Model Composer 2024.1

Hardware

- Architecture: Versal adaptive SoCs

After completing this comprehensive training, you will have the necessary skills to:

- Describe the AMD Versal AI Engine architecture
- Utilize the AI Engine DSP library and create a filter design with the AMD Vitis Unified IDE
- Follow the system partitioning and system mapping methodology
- Add custom kernel code to a design
- Design a DSP function with the Vitis Model Composer AI Engine library
- Analyze AI Engine designs using the Vitis Analyzer utility

Course Outline

Day 1

- **AMD Versal AI Engine Architecture**
  Introduces the architecture of the AI Engine and its components. {Lecture}
- **Introduction to the AI Engine DSP Library**
  Provides an overview of the AI Engine DSP library, which enables faster development and comes with ready-to-use example design that help with using the library and tools. {Lecture, Labs}
- **System Partitioning Methodology**
  Covers the system design planning and partitioning methodology for mapping design requirements to the AI Engine. {Lecture, Lab}
- **Rapid Prototyping and Custom Coding of AI Engine Kernels**
  Describes the AI Engine programming flow with kernels and Adaptive Data Flow (ADF) graphs. Also outlines the kernel coding methodology for writing custom kernel code and rapid prototyping. {Lecture, Lab}
- **Overview of AI Engine Kernel Optimization**
  Highlights the various AI Engine kernel optimization techniques, such as compiler directives, software pipelining, coding for performance, and core utilization. {Lecture}
- **Analyzing AI Engine Designs Using the Vitis Analyzer**
  Covers the different reports generated by the Vitis Unified IDE and how to use these reports to optimize and debug AI Engine kernels. {Lecture}
- **AI Engine DSP Designs with Vitis Model Composer**
  Describes the Vitis Model Composer tool and how to use the libraries available with the tool for AI Engine DSP design development. {Lecture, Lab}